

## **REMARKS**

### **Introduction**

Claims 1, 3, 5-8, 11, 13, 15-18 and 21 have been amended. Claims 2, 4, 12, 14 and 25 have been cancelled. The application now includes claims 1, 3, 5-11, 13 and 15-21. Reconsideration of the rejection of the application is respectfully requested in view of the claim amendments and the following remarks.

An RCE is being filed herewith.

### **The Claims are Allowable at least because the Prior Art Fails to Disclose Cell-Based Data Processing in which a Specification includes a Plurality of Cells**

Claims 1-21 and 25 are rejected under 35 U.S.C. § 103(a) as being unpatentable over “XML Path language (XPath)” and “XSL Transformation (XSLT) Version 1.0 (W3C Recommendation 16 November 1999) in view of Renner et al. (U.S. Patent No. 6,993,657, “Renner”). Reconsideration of this rejection is respectfully requested at least because the prior art fails to disclose cell-based data processing in which a specification includes a plurality of cells.

One embodiment of the present invention is a computer-implemented method of cell-based data processing. U.S. Pat. Pub. No. 2002/0133808 (publication of the present application), Abstract. A data processing specification or “x-sheet” 102 includes a plurality of cells or “x-cells” 104. See *id.* at ¶ [0019]. Each cell includes a formula specifying an action or computation to perform when the cell is executed, and one or more attributes referencing other cells. See *id.* Further, each cell is delineated by a beginning and ending tag, and one of the cells is reserved as an output cell for

outputting a result of the processing. *See id.* at ¶ [0023]. An execution analyzer 122 parses the specification to determine an interdependency of the plurality of cells, and generates and stores a directed graph of the interdependency as an execution flow. *See id.* at ¶¶ [0068] - [0069]. An execution engine 124 then executes the specification in accordance with the execution flow by evaluating the formula of each cell in the execution flow. *See id.* at ¶ [0072].

XSLT generally discusses defining “the syntax and semantics of the XSLT language” which is “a stylesheet language for XML” (see Abstract, second paragraph, and section 1, first paragraph). “A transformation expressed in XSLT describes rules for transforming a source tree into a result tree. The transformation is achieved by associating patterns with templates” (section 1, second paragraph, of XSLT). “A stylesheet contains a set of template rules. A template rule has two parts: a pattern which is matched against nodes in the source tree and a template which can be instantiated to form part of the result tree” (section 1, fifth paragraph, of XSLT). “[T]he `xsl:value-of` element can be used to compute generated text, for example by extracting text from the source tree or by inserting the value of a variable. The `xsl:value-of` element does this with an expression that is specified as the value of the `select` attribute” (section 7.6, first paragraph, of XSLT). In other words, XSLT searches for patterns in nodes in a source tree and applies templates thereto. For example, the template:

```
<xsl:template match="person">
  <p>
    <xsl:value-of select="@given-name"/>
    <xsl:text> </xsl:text>
```

```
<xsl:value-of select="@family-name"/>
</p>
</xsl:template>
```

“creates an HTML paragraph from a person element with given-name and family-name attributes” (see section 7.6.1 of XSLT).

On the other hand, the amended claims recite generating execution flow descriptions based on interaction or computation references between data processing cells. This differs from XSLT, which merely searches for patterns in nodes in a source tree and applies templates thereto, in essence performing a “find-and-replace” operation. XSLT is completely silent as to determining an execution flow where interdependencies exist among cells. Further, nothing is cited or found in XPath or Renner et al. or any other cited prior art overcomes these deficiencies of XSLT.

In contrast to the cited art, amended independent claim 1 recites a specification that includes cells and that “generates and stores a directed graph of the interdependency as an execution flow” from the cells. For at least the reasons discussed above, independent claim 1, and independent claims 11 and 21, which recite similar features but have a different scope, are allowable over the cited art. The remaining claims depend from independent claims 1 or 11 and add further features thereto. Accordingly, the dependent claims are also allowable over the cited art for at least the reasons discussed above with respect to the independent claims.

**Conclusion**

Applicants respectfully request favorable action in connection with this application.

The Examiner is invited and urged to contact the undersigned to discuss any matter concerning this application.

A one month extension of time should be required for this submission. Should any other fee be required, the Commissioner is authorized to charge any such fee to Counsel's Deposit Account 50-2222.

Respectfully submitted,

Date: January 19, 2010

/Barry S. Goldsmith/

Barry S. Goldsmith  
Attorney for Applicants  
Registration No. 39,690

**Customer No. 74739**  
SQUIRE, SANDERS & DEMPSEY LLP  
14<sup>TH</sup> Floor  
8000 Towers Crescent Drive  
Vienna, Virginia 22182-6212  
Telephone: 703-720-7876  
Fax: 703-720-7802